ID: 221-1552

NAME: Syed Arhan Ahmed SECTION: CY-B

Read the Instructions Carefully

DUE DATE: 30th March, 2024 11:59 PM

Print this assignment and complete it on the printed sheet only. No extra sheet is allowed. Scan/Camscan the solution and upload it on Google Classroom.

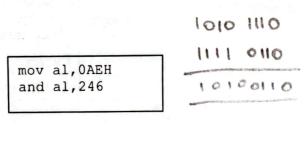
You can verify the answers using MASM debugging mode.

Cheating in any case if found will be marked as ZERO in whole assignment category.

Perform all the steps in the "calculation" box, only filling the answer will not get any credit.

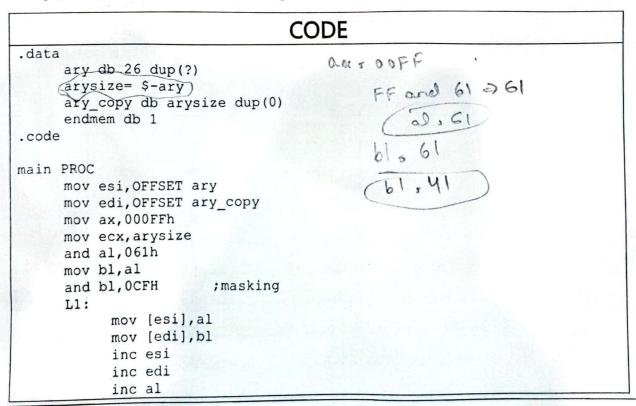
1. Update Flags after executing following code?

NOTE: AND performs a Boolean(bitwise) AND operation between each pair of matching bits in two operands and place result in the destination. AND instruction always clear overflow and carry flags. It modifies Sign, Zero and Parity flags.



	Sign	1	
	Zero	0	
Tela ma	Carry	0	Tcleared
Flags	Overflow	0	
	Parity	1	
	Auxiliary	0]un-change

2. Update memory after executing code given below



inc bl LOOP L1 INVOKE ExitProcess, 0 main ENDP END main **MEMORY** C E 0 1 2 3 5 6 7 8 D 68 69 63 67 6A 65 66 68 6F 70 61 64 4000 62 6D 6c 43 45 46. 42 44 4010 72 73 74 76 78 41 52 48 40 4020 us MA YE 44 47 50 UD 58 59 5A 01

3. Update Flags after executing following code

NOTE: OR performs a Boolean(bitwise) OR operation between each pair of matching bits in two operands and place result in the destination. OR instruction always clear overflow and carry flags. It modifies Sign, Zero and Parity flags.

and the second second second			11100011
mov al,11100011b OR al,00000100b;	setting		11100111
OR al,00000100b;	setting	3rd bit	11100111

	Sign	1	
	Zero	0	
Flags	Carry	0	1
Flags	Overflow	0	
	Parity	61	
	Auxiliary	0	•

4. Update memory after executing code given below

en below 111 o om

```
CODE
.data
                                  nerroy filled
                                                  Spio of
     ary db 26 dup(?)
                                                  yter:
                                   allowed
     arysize= $-ary
     ary copy db arysize dup(0)
     endmem db 1
.code
                                       11400
    mov esi, OFFSET ary
    mov edi, OFFSET ary copy
                                 61 = 41h
    mov ax, 0041h
    mov ecx, arysize
                                          0110 0001 = 611
    mov bl,al
    OR bl,00100000b
    L1:
          mov [esi], al
          mov [edi],bl
          inc esi
          inc edi
          inc al
          inc bl
    LOOP L1
```

N.	4		RY
IV		V	Γ

	0	1	2	3	4	5	6	7	8	9	Α	В	C	D	E	F
4000	41	42	43	44	45	46	47	48	49	4A	48	40	40	4E	45	50
4010	CI	52	5.3	54	SS	56	37	58	59	SA	61	62	63	64	65	66
4020	67	68	69		68	60	eD.	6E	6F	70	71	72	73	74	75	7.6

5. Update register after each line of code and update Flags after executing the following code

1111 1111 1111

	7.42.000		
mov ax, 0A593H		Sign	
XOR ax, -1		Sign	0
XOR ax, 0		Zero	0
XOR ax,-1		C	6
XOR ax,ax	Flags	Carry	0
mov ax, 05A37H	riags	Overflow	0
XOR al,0		Parity	1
XOR ah,1		Auxiliary	0
XOR al, ah			0110
1		1 16 -	0110

AX	AS	93
AX	SA	93 6C 6C
AX	SA	60
AX	A5	93
AX	00	00
AX Al	5A	37
	5 A 3	37 7

SA XOR D SB 37xOR SB

6. Write a program that finds parity of number given below? HINT: Use XOR and LOOP to find parity

.data
parity DQ 0A1B2C3D4E5F67890H

Answer:

dosseg

model small

main proc

moved an effective

stack look

move are, adama

paint Da OAIB2C3DHESF67890H moved, byte ptr paint

even alb 10,13, Even paints' Koll al, a

page 3 of 7 moved are, offset even main end p

7.	Update flags afte	r arithmetic	instructio	? Also state which of the following jumps will taken or not
	taken	an,	FF FE	FFFE + FE 70 DIII 1100 01101110

.code

mov ax,0FFFEh
add ax,0FC70h
jc 11
L1: jz L2
L2: jo L3
L3: js L4
L4: jp L5

L5:

	aas FCGE	, 0
	TAKEN	NOT TAKEN
Jc	/	
Jz		

	Sign	1		you	1-(1-1		alculation	
	Zero	0		IIII	iiii	1111	(110	* 1000 - 120 - 110
Flags	Carry	1	1 (+ 1111	1100	0111	0000	
C	Overflow	0	1 }	01111	1100	0110	1110	
	Parity	0	1					

Jo

Js

jp

8. Update flags after arithmetic instruction? Also state which of the following jumps will taken or not taken

.code			
		m	nov ax,07B1Ah
		S	sub ax, OCEEBh
		j	c 11
	L1:	jz	L2
	L2:	jo	L3
	L3:	js	L4
	L4:	qį	L5
	7 C.		

	TAKEN	NOT TAKEN
Jc		
Jz		
Jo		
Js	-	
jp		/

	Sign	1	Calculation
	Zero	0	0101 1011 0001 1010
Flags	Carry	1	0011 0001 0001 0101
	Overflow	1	1010 1100 0010 1111
	Parity	0	

7BIA - CEEB an, 7BIA

+ 1 0001 0001 0100 Page 4 of 7

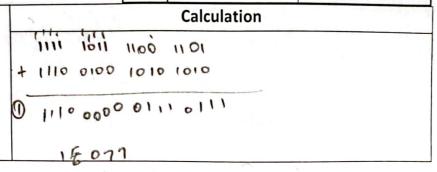
+ 1 0001 0001 0001 0101

9. Update flags after arithmetic instruction? Also state which of the following jumps will taken or not taken

mov cx,0FBCDh
add cx,0E4AAh
jnc 11
L1: jnz L2
L2: jo L3
L3: js L4
L4: jnp L5

	TAKEN	NOT TAKEN
Jnc		/
Jnz		V
Jo		
Js		
jnp		~

300	Sign	1
	Zero	0
	Carry	1
Flags	Overflow	0
	Parity	١
	Auxiliary	١



10. Update flags after arithmetic instruction? Also state which of the following jumps will taken or not taken

add bx,0684Ah jc l1 L1: jz L2 L2: jno L3 L3: jns L4 L4: jp L5 L5: mov ah,04ch

mov bx, 0FABDh

	TAKEN	NOT TAKEN
Jc	1	
Jz	724	/
Jno		
Jns	/	
jp	When the state of	/

	Sign	0
	Zero	0
	Carry	1
Flags	Overflow	0
	Parity	0
	Auxiliary	1

1118			Calculation	
1111	1010	1011	1011	
+011	0 1000	0100	1010	
	A manager agreement	AND DESCRIPTION OF THE PERSON	1011	

11. Update value of ax and cx registers after every iteration. Update any changes to the done to flag

mov ecx,5 mov ax,1 L1: inc ax dec ecx jcxz end_loop jmp L1 end loop:

	1	2	3	4	5	6
сх	05	04	03	02	01	00
AX	01	02	03	04	05	06

ECX = 0

	Sign	0	Calculation
	Zero	1	ECX is = 0 in
T)	Carry	0	
Flags	Overflow	0	The end.
	Parity	1	
	Auxiliary	0	

12. Fill flag after every CMP instruction

mov al, +127

CMP al, -128

jg IsGreater

ja IsAbove

NOT TAKEN **TAKEN** Ja Jg

FF als 7F QE - (-128) 1000 0000 Till IIIO

Calculation @ Sign 1111 1110 Zero 0 1000 0000 0 Carry 1111 (111 **Flags** Overflow 0 255 > 128 **Parity** 1 Auxiliary

mov dx,-1 CMP dx,

jnl L5 jnle L5 j1 L1

	TAKEN	NOT TAKEN
Jnl		1
Jnle		1
JL	/	4

	Sign	1	Calculation	
	Zero	0	1111 1111 1111 111	
***	Carry	0	- 1111 1111 1111	
Flags	Overflow	0		
	Parity	1	1414 44 44 1114	
	Auxiliary	0	100 1111 100	

dus FF FF

mov bx, +32 cmp bx, -35

jng L5 jnge L5 jge L1

	TAKEN	NOT TAKEN
JNG		/
JNGE		/
JGE	/	

	Sign	0	Calculation
	Zero	0	0010 0000
	Carry	1	0010 0014
Flags	Overflow	0	01000010
	Parity	0	
	Auxiliary	O	0100 001
	-		67

mov cx,0

jg L5 jnl L1 jle L2

		NOT
1- 11	TAKEN	TAKEN
JG		/
JNL	1	Ser.
JLE	/	19

Flags	Sign	0	Calculation
	Zero	()	0000000
	Carry	0	
	Overflow	0	
	Parity	1	
	Auxiliary	0	

mov cx,0

jl L5 jng L1 jge L2

		NOT
	TAKEN	TAKEN
JL		
JNG	1	/ .
JGE	1	

Flags	Sign	0	Calculation
	Zero	1	0000000
	Carry	0	0000000
	Overflow	0	FOOODNOO